Nonmonotonic Probabilistic Logics under Variable-Strength Inheritance with Overriding: Algorithms and Implementation in NMPROBLOG

Thomas Lukasiewicz*
Dipartimento di Informatica e Sistemistica
Università degli Studi di Roma "La Sapienza"
Via Salaria 113, I-00198 Rome, Italy
lukasiewicz@dis.uniroma1.it

Abstract

In previous work, I have introduced nonmonotonic probabilistic logics under variable-strength inheritance with overriding. They are formalisms for probabilistic reasoning from sets of strict logical, default logical, and default probabilistic sentences, which are parameterized through a value $\lambda \in [0,1]$ that describes the strength of the inheritance of default probabilistic knowledge. In this paper, I continue this line of research. I present algorithms for deciding consistency of strength λ and for computing tight consequences of strength λ, which are based on reductions to the standard problems of deciding satisfiability and of computing tight logical consequences in model-theoretic probabilistic logic. Furthermore, I describe an implementation of these algorithms in the system NMPROBLOG.

Keywords. Model-theoretic and nonmonotonic probabilistic logics, qualitative reasoning about uncertainty, probability rankings, algorithms for manipulating imprecise probabilities, convex sets of probability measures.

1 Introduction

During the recent decades, reasoning about probabilities has started to play an important role in artificial intelligence (AI). In particular, reasoning about interval restrictions for conditional probabilities, also called conditional constraints [28], has been a subject of extensive research efforts. Roughly, a conditional constraint is of the form $(\psi|\phi)[l,u]$, where ψ and ϕ are events, and $[l,u]$ is a subinterval of the unit interval $[0,1]$. It encodes that the conditional probability of ψ given ϕ lies in $[l,u]$.

An important approach for handling conditional constraints is model-theoretic probabilistic logic, which has its origin in philosophy and logic, and whose roots can be traced back to already Boole in 1854 [10]. There is a wide spectrum of formal languages that have been explored in model-theoretic probabilistic logic, ranging from constraints for unconditional and conditional events to rich languages that specify linear inequalities over events (see especially the work by Nilsson [33], Fagin et al. [14], Dubois and Prade et al. [11, 12], Frisch and Haddawy [16], and the author [27, 28]). The main decision and optimization problems in probabilistic logic are deciding satisfiability, deciding logical consequence, and computing tight logically entailed intervals.

For example, a simple collection of conditional constraints KB may encode the *strict logical knowledge* "all eagles are birds" and "all birds have feathers" as well as the *purely probabilistic knowledge* "birds fly with a probability of at least 0.95". This KB is satisfiable, and some logical consequences in model-theoretic probabilistic logic from KB are "all birds have feathers", "birds fly with a probability of at least 0.95", "all eagles have feathers", and "eagles fly with a probability between 0 and 1"; in fact, these are the tightest intervals that follow from KB. That is, we especially cannot conclude anything from KB about the ability to fly of eagles.

A closely related research area is default reasoning from conditional knowledge bases, which consist of a collection of strict statements in classical logic and a collection of defeasible rules, also called defaults. The former must always hold, while the latter are rules of the kind $\psi \leftarrow \phi$, which read as "generally, if ϕ then ψ." Such rules may have exceptions, which can be handled in different ways.

The literature contains several different proposals for default reasoning from conditional knowledge bases and extensive work on its desired properties. The core of these properties are the rationality postulates of System P by Kraus, Lehmann, and Magidor [22], which constitute a sound and complete axiom system for several model-theoretic entailment relations under uncertainty measures on worlds. They characterize model-theoretic entailment under preferential structures, infinitesimal probabilities, possibility measures, and world rankings. As shown by Friedman and Halpern [15], many of these uncertainty measures on worlds are expressible as plausibility mea-

* Alternate address: Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, A-1040 Vienna, Austria; e-mail: lukasiewicz@kr.tuwien.ac.at.

sures. See [5, 17] for a survey of the above relationships.

Mainly to solve problems with irrelevant information, the notion of rational closure as a more adventurous notion of entailment was introduced by Lehmann [25]. It is equivalent to entailment in System Z by Pearl [35] and to the least specific possibility entailment by Benferhat et al. [4]. Finally, mainly to solve problems with property inheritance from classes to exceptional subclasses, some more sophisticated notions of entailment were proposed, including the notion of lexicographic entailment by Lehmann [26] and Benferhat et al. [3].

For example, a conditional knowledge base KB may encode the *strict logical knowledge* "all ostriches are birds" and the *default logical knowledge* "generally, birds fly", "generally, ostriches do not fly", and "generally, birds have wings". Some desirable conclusions from KB [21] are "generally, birds fly" and "generally, birds have wings" (which both belong to KB), "generally, ostriches have wings" (since the set of all ostriches is a subclass of the set of all birds, and thus ostriches should inherit all properties of birds), "generally, ostriches do not fly" (since properties of more specific classes should override inherited properties of less specific classes), and "generally, red birds fly" (since "red" is not mentioned at all in KB and thus should be irrelevant to the ability to fly of birds).

There are several works in the literature on probabilistic foundations for default reasoning from conditional knowledge bases [1, 34, 9], on combinations of Reiter's default logic with statistical inference [24, 37], and on a rich first-order formalism for deriving degrees of belief from statistical knowledge including default statements [2]. A series of recent papers has proposed combinations of model-theoretic probabilistic reasoning from conditional constraints with default reasoning from conditional knowledge bases, which are summarized as follows:

• The paper [32] presents *weak nonmonotonic probabilistic logics*, which are extensions of probabilistic logic by defaults as in conditional knowledge bases under Kraus et al.'s System P [22], Pearl's System Z [35], and Lehmann's lexicographic entailment [26]. The new formalisms allow for expressing in a uniform framework *strict logical knowledge* and *purely probabilistic knowledge* from probabilistic logic, as well as *default logical knowledge* from default reasoning from conditional knowledge bases. For example, consider the *strict logical knowledge* "all penguins are birds", the *default logical knowledge* "generally, birds have legs" and "generally, birds fly", and the *purely probabilistic knowledge* "penguins fly with a probability of at most 0.05". Obviously, some desired conclusions are "generally, birds have legs", "generally, birds fly", and "penguins fly with a probability of at most 0.05", since these sentences are explicitly stated above. Two other desired conclusions are "generally, penguins have legs" (since the property of having legs of birds should be inherited down to the subclass of all penguins) and "generally, red birds fly" (since the property of being red is not mentioned at all above, and thus should be irrelevant to the ability to fly). In weak nonmonotonic probabilistic logics, we can deal with all the above sentences. In particular, the notion of probabilistic lexicographic entailment also produces all the above desired conclusions.

• A companion paper [30] presents *strong nonmonotonic probabilistic logics*, which are similar probabilistic generalizations of default reasoning from conditional knowledge bases. They are, however, quite different from the ones in [32] in that they allow for handling *default purely probabilistic knowledge*, rather than *(strict) purely probabilistic knowledge*, in addition to strict logical knowledge and default logical knowledge. For example, they allow for expressing sentences "*generally*, birds fly with a probability of at least 0.95" rather than "birds fly with a probability of at least 0.95". Intuitively, the former means that being able to fly with a probability of at least 0.95 should apply to all birds and all subclasses of birds, as long as this is consistent, while the latter says that being able to fly with a probability of at least 0.95 should only apply to all birds. This is why the formalisms in [30] are generally much stronger than the ones in [32].

• Finally, the papers [31] define *nonmonotonic probabilistic logics under variable-strength inheritance with overriding*, which are a general approach to nonmonotonic probabilistic reasoning, which subsumes the approaches in [32] and [30] as special cases. Roughly, these formalisms also allow for handling *strict logical knowledge*, *default logical knowledge*, and *default purely probabilistic knowledge*, but the inheritance of purely probabilistic knowledge is controlled by a strength $\lambda \in [0,1]$. For $\lambda = 0$ (resp., $\lambda = 1$), these formalisms coincide with the weak (resp., strong) nonmonotonic ones in [32] (resp., [30]). For example, suppose that we have the default probabilistic knowledge "generally, yellow objects are easy to see with a probability between 0.8 and 0.9". In nonmonotonic probabilistic reasoning of strength 0 (resp., 0.5 and 1), we then conclude "generally, yellow birds are easy to see with a probability in $[0,1]$ (resp., $[0.6,1]$ and $[0.8,0.9]$)".

To date, however, there have been no algorithms for nonmonotonic probabilistic logics under variable-strength inheritance with overriding. Furthermore, there have been no implementations, neither of these unifying formalisms, nor of the special cases of weak and strong nonmonotonic probabilistic logics. In this paper, I try to fill this gap. The main contributions can be summarized as follows:

• I recall the nonmonotonic probabilistic logics under variable-strength inheritance with overriding presented in [31], namely, *probabilistic entailment in System Z of strength λ* (or *z_λ-entailment*) and *probabilistic lexicographic entailment of strength λ* (or *lex_λ-entailment*).

I also provide several new examples.

• I present an algorithm for deciding consistency of strength λ, which is based on a reduction to deciding satisfiability in model-theoretic probabilistic logic. I also present algorithms for computing tight entailed intervals under z_λ- and lex_λ-entailment, based on reductions to deciding satisfiability and computing tight logically entailed intervals in model-theoretic probabilistic logic.

• I describe the system NMPROBLOG, which includes implementations of these algorithms. Deciding satisfiability (resp., computing tight logically entailed intervals) in model-theoretic probabilistic logic are reduced to deciding the solvability of a system of linear constraints (resp., solving linear programs), which is done by "lp_solve".

The rest of this paper is organized as follows. Section 2 gives some technical preliminaries. In Sections 3 and 4, we recall the notions of z_λ-and lex_λ-entailment, and their semantic properties. In Section 5, we give some further examples to illustrate the notions of z_λ- and lex_λ-entailment. Sections 6 describe the algorithms for deciding consistency of strength λ and computing tight entailed intervals under z_λ- and lex_λ-entailment. In Section 7, we present the system NMPROBLOG. Section 8 summarizes the main results and gives an outlook on future research.

2 Preliminaries

In this section, we recall probabilistic knowledge bases and the main concepts from model-theoretic probabilistic logic. Furthermore, we define the monotonic notion of logical entailment of strength $\lambda \in [0,1]$.

Probabilistic Knowledge Bases. We now recall probabilistic knowledge bases. We start by defining logical constraints and probabilistic formulas, which are interpreted by probability distributions over a set of possible worlds.

We assume a set of *basic events* $\Phi = \{p_1, \ldots, p_n\}$ with $n \geqslant 1$. We use $\bot$ and $\top$ to denote *false* and *true*, respectively. We define *events* by induction as follows. Every element of $\Phi \cup \{\bot, \top\}$ is an event. If ϕ and ψ are events, then also $\neg\phi$ and $(\phi \wedge \psi)$. A *conditional event* is an expression of the form $\psi|\phi$, where ψ and ϕ are events. A *conditional constraint* has the form $(\psi|\phi)[l,u]$, where ψ and ϕ are events, and $l,u \in [0,1]$ are reals. We define *probabilistic formulas* by induction as follows. Every conditional constraint is a probabilistic formula. If F and G are probabilistic formulas, then also $\neg F$ and $(F \wedge G)$. We use $(F \vee G)$ and $(F \Leftarrow G)$ to abbreviate $\neg(\neg F \wedge \neg G)$ and $\neg(\neg F \wedge G)$, respectively, where F and G are either two events or two probabilistic formulas, and adopt the usual conventions to eliminate parentheses. A *logical constraint* is an event of the form $\psi \Leftarrow \phi$.

A *world* I is a truth assignment to the basic events in Φ (that is, a mapping $I\colon \Phi \to \{\mathbf{true}, \mathbf{false}\}$), which is inductively extended to all events as usual (that is, $I(\bot) = \mathbf{false}$, $I(\top) = \mathbf{true}$, $I(\neg\phi) = \mathbf{true}$ iff $I(\phi) = \mathbf{false}$, and $I(\phi \wedge \psi) = \mathbf{true}$ iff $I(\phi) = I(\psi) = \mathbf{true}$). We denote by $\mathcal{I}_\Phi$ the set of all worlds for Φ. A world I *satisfies* an event ϕ, or I is a *model* of ϕ, denoted $I \models \phi$, iff $I(\phi) = \mathbf{true}$. A *probabilistic interpretation* Pr is a probability function on $\mathcal{I}_\Phi$ (that is, a mapping $Pr\colon \mathcal{I}_\Phi \to [0,1]$ such that all $Pr(I)$ with $I \in \mathcal{I}_\Phi$ sum up to 1). The *probability* of an event ϕ in Pr, denoted $Pr(\phi)$, is the sum of all $Pr(I)$ such that $I \in \mathcal{I}_\Phi$ and $I \models \phi$. For events ϕ and ψ with $Pr(\phi) > 0$, we write $Pr(\psi|\phi)$ to abbreviate $Pr(\psi \wedge \phi) \,/\, Pr(\phi)$. The *truth* of logical constraints and probabilistic formulas F in Pr, denoted $Pr \models F$, is inductively defined by: (i) $Pr \models \psi \Leftarrow \phi$ iff $Pr(\psi \wedge \phi) = Pr(\phi)$, (ii) $Pr \models (\psi|\phi)[l,u]$ iff $Pr(\phi) = 0$ or $Pr(\psi|\phi) \in [l,u]$, (iii) $Pr \models \neg F$ iff not $Pr \models F$, and (iv) $Pr \models (F \wedge G)$ iff $Pr \models F$ and $Pr \models G$. We say Pr *satisfies* F, or Pr is a *model* of F, iff $Pr \models F$. It *satisfies* a set of logical constraints and probabilistic formulas $\mathcal{F}$, or Pr is a *model* of $\mathcal{F}$, denoted $Pr \models \mathcal{F}$, iff Pr is a model of all $F \in \mathcal{F}$. We say $\mathcal{F}$ is *satisfiable* iff a model of $\mathcal{F}$ exists. A conditional constraint $C = (\psi|\phi)[l,u]$ is a *logical consequence* of $\mathcal{F}$, denoted $\mathcal{F} \Vdash C$, iff each model of $\mathcal{F}$ is also a model of C. It is a *tight logical consequence* of $\mathcal{F}$, denoted $\mathcal{F} \Vdash_{tight} C$, iff $l = \inf Pr(\psi|\phi)$ (resp., $u = \sup Pr(\psi|\phi)$) subject to all models Pr of $\mathcal{F}$ with $Pr(\phi) > 0$. Here, we define $l = 1$ and $u = 0$, when no such model Pr exists.

A *probabilistic knowledge base* $KB = (L, P)$ consists of a finite set of logical constraints L and a finite set of conditional constraints P. We say KB is *satisfiable* iff $L \cup P$ is satisfiable. A conditional constraint C is a *logical consequence* of KB, denoted $KB \Vdash C$, iff $L \cup P \Vdash C$. It is a *tight logical consequence* of KB, denoted $KB \Vdash_{tight} C$, iff $L \cup P \Vdash_{tight} C$. The following example illustrates the syntactic notion of a probabilistic knowledge base.

Example 2.1 The strict logical knowledge "all penguins are birds", the default logical knowledge "generally, birds have legs", and the default purely probabilistic knowledge "generally, yellow objects are easy to see with a probability between 0.8 and 0.9", "generally, birds fly with a probability of at least 0.9", and "generally, penguins fly with a probability of at most 0.1" can be expressed by the probabilistic knowledge base $KB = (L, P)$, where $L = \{\mathit{bird} \Leftarrow \mathit{penguin}\}$ and $P = \{(\mathit{legs}|\mathit{bird})[1,1]$, $(\mathit{see}|\mathit{yellow})[.8,.9]$, $(\mathit{fly}|\mathit{bird})[.9,1]$, $(\mathit{fly}|\mathit{penguin})[0,.1]\}$.

Logical Entailment of Strength λ. As a first step towards z_λ- and lex_λ-entailment in Section 3, we now define the monotonic notion of *logical entailment of strength* $\lambda \in [0,1]$. It already realizes an inheritance of default purely probabilistic knowledge controlled by λ. But, in contrast to z_λ- and lex_λ-entailment, it has no overriding mechanism, and this often produces *local inconsistencies*.

In the sequel, we use $\phi \succcurlyeq \lambda$ to abbreviate the probabilistic formula $\neg(\phi|\top)[0,0] \wedge (\phi|\top)[\lambda,1]$. Informally, for any probabilistic interpretation Pr that satisfies $\phi \succcurlyeq \lambda$, it holds that $Pr(\phi) > 0$, if $\lambda = 0$, and $Pr(\phi) \geqslant \lambda$, otherwise. We define the notion of *logical entailment of strength* $\lambda \in [0,1]$ (or simply λ-*logical entailment*) as follows. A conditional constraint $C = (\psi|\phi)[l,u]$ is a λ-*logical consequence* of $KB = (L,P)$, denoted $KB \Vdash^{\lambda} C$, iff $L \cup P \cup \{\phi \succcurlyeq \lambda\} \Vdash C$. It is a *tight* λ-*logical consequence* of KB, denoted $KB \Vdash^{\lambda}_{tight} C$, iff $L \cup P \cup \{\phi \succcurlyeq \lambda\} \Vdash_{tight} C$.

Example 2.2 Let KB be as in Example 2.1. Some tight logical consequences of strength λ among 0, 0.2, 0.4, 0.6, 0.8, and 1 are shown in Table 1 (less desired intervals are bold). We observe an inheritance of default logical knowledge along subclass relationships, which is independent from λ. E.g., the default logical property of having legs is inherited from birds down to yellow birds. Furthermore, we observe an inheritance of default purely probabilistic knowledge along subclass relationships, which depends on the strength λ. E.g., being easy to see with a probability in $[.8,.9]$ is inherited from yellow objects down to yellow birds, but the new intervals are $[0,1]$, $[0,1]$, $[.5,1]$, $[.67,1]$, $[.75,1]$, and $[.8,.9]$, respectively. Finally, for $\lambda > 1/9$, there are local inconsistencies related to penguins (as being able to fly with a probability of at least 0.9 is inherited from birds down to penguins, and there it is inconsistent with being able to fly with a probability of at most 0.1).

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$legs\|bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\|yellow \wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\|penguin$	$[1,1]$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$
$legs\|yellow \wedge penguin$	$[1,1]$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$
$fly\|bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly\|yellow \wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly\|penguin$	$[0,.1]$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$
$fly\|yellow \wedge penguin$	$[0,1]$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$
$see\|yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see\|yellow \wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see\|yellow \wedge penguin$	$[0,1]$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$

Table 1: Some tight λ-logical consequences.

3 Nonmonotonic Probabilistic Logics

In this section, we recall the notions of z_λ- and lex_λ-entailment from [31]. They are parameterized through a value $\lambda \in [0,1]$ that describes the *strength* of the inheritance of default purely probabilistic knowledge.

Consistency of Strength λ. We now recall the notion of consistency of strength λ (or λ-consistency) for probabilistic knowledge bases $KB = (L,P)$.

A probabilistic interpretation Pr λ-*verifies* (resp., λ-*falsifies*) a conditional constraint $(\psi|\phi)[l,u]$ iff Pr verifies (resp., falsifies) $(\psi|\phi)[l,u]$ and satisfies $\phi \succcurlyeq \lambda$. A set of conditional constraints P λ-*tolerates* a conditional constraint C under a set of logical constraints L iff $L \cup P$ has a model that λ-verifies C. We say P is under L *in* λ-*conflict* with C iff no model of $L \cup P$ λ-verifies C. A conditional constraint ranking σ on $KB = (L,P)$ is λ-*admissible* with KB iff every $P' \subseteq P$ that is under L in λ-conflict with some $C \in P$ contains some C' such that $\sigma(C') < \sigma(C)$.

We are now ready to define the notion of λ-consistency. We say KB is λ-*consistent* iff there exists a conditional constraint ranking σ on KB that is λ-admissible with KB. The following theorem characterizes the λ-consistency of KB through the existence of an ordered partition of P.

Theorem 3.1 *A probabilistic knowledge base $KB = (L, P)$ is λ-consistent iff an ordered partition $(P_0, \ldots, P_k)$ of P exists such that each P_i, $0 \leqslant i \leqslant k$, is the set of all $C \in \bigcup_{j=i}^{k} P_j$ that are λ-tolerated under L by $\bigcup_{j=i}^{k} P_j$.*

The unique ordered partition $(P_0, \ldots, P_k)$ of P in Theorem 3.1 is called the z_λ-*partition* of $KB = (L,P)$. Hence, KB is λ-consistent iff its z_λ-partition exists. The following example shows some z_λ-partitions.

Example 3.2 Let $KB = (L,P)$ be as in Example 2.1. For every $\lambda \in [0,1/9]$, the z_λ-partition is given by $(P_0) = (P)$. For every $\lambda \in (1/9,1]$, the z_λ-partition is given by $(P_0,P_1) = (P \setminus \{(fly|penguin)[0,.1]\}, \{(fly|penguin)[0,.1]\})$. Thus, KB is λ-consistent, for all $\lambda \in [0,1]$.

It can also be shown that $KB = (L,P)$ is λ-consistent iff a probability ranking κ exists that is λ-admissible with KB. Formally, a *probability ranking* κ maps each probabilistic interpretation on $\mathcal{I}_\Phi$ to a member of $\{0,1,\ldots\} \cup \{\infty\}$ such that $\kappa(Pr) = 0$ for at least one interpretation Pr. It is extended to all logical constraints and probabilistic formulas F as follows. If F is satisfiable, then $\kappa(F) = \min\{\kappa(Pr) \mid Pr \models F\}$; otherwise, $\kappa(F) = \infty$. A probability ranking κ is λ-*admissible* with a probabilistic knowledge base $KB = (L,P)$ iff $\kappa(\neg F) = \infty$ for all $F \in L$ and $\kappa(\phi \succcurlyeq \lambda) < \infty$ and $\kappa(\phi \succcurlyeq \lambda \wedge (\psi|\phi)[l,u]) < \kappa(\phi \succcurlyeq \lambda \wedge \neg(\psi|\phi)[l,u])$ for all $(\psi|\phi)[l,u] \in P$.

System Z of Strength λ. We next recall the notion of z_λ-entailment, $\lambda \in [0,1]$, for λ-consistent $KB = (L,P)$.

It is linked to a conditional constraint ranking z_λ on KB and a probability ranking κ^{z_λ}. Let $(P_0, \ldots, P_k)$ be the z_λ-*partition* of KB. For every $j \in \{0, \ldots, k\}$, each $C \in P_j$ is assigned the value j under z_λ. Then, κ^{z_λ} on all probabilistic interpretations Pr is defined as follows:

$$\kappa^{z_\lambda}(Pr) = \begin{cases} \infty & \text{if } Pr \not\models L \\ 0 & \text{if } Pr \models L \cup P \\ 1 + \max\limits_{C \in P:\ Pr \not\models C} z_\lambda(C) & \text{otherwise.} \end{cases}$$

The probability ranking κ^{z_λ} defines a preference relation on probabilistic interpretations: For probabilistic interpretations Pr and Pr', we say Pr is z_λ***-preferable*** to Pr' iff $\kappa^{z_\lambda}(Pr) < \kappa^{z_\lambda}(Pr')$. A model Pr of a set of logical constraints and probabilistic formulas $\mathcal{F}$ is a z_λ***-minimal model*** of $\mathcal{F}$ iff no model of $\mathcal{F}$ is z_λ-preferable to Pr.

We are now ready to define the notion of z_λ***-entailment*** as follows. A conditional constraint $C = (\psi|\phi)[l, u]$ is a z_λ***-consequence*** of KB, denoted $KB \mathrel{|\!\!\sim}^{z_\lambda} C$, iff every z_λ-minimal model of $L \cup \{\phi \succcurlyeq \lambda\}$ satisfies C. It is a ***tight*** z_λ***-consequence*** of KB, denoted $KB \mathrel{|\!\!\sim}^{z_\lambda}_{tight} C$, iff $l = \inf Pr(\psi|\phi)$ (resp., $u = \sup Pr(\psi|\phi)$) subject to all z_λ-minimal models Pr of $L \cup \{\phi \succcurlyeq \lambda\}$.

The following example shows that the notion of z_λ-entailment realizes an inheritance of default logical and default purely probabilistic properties from classes to non-exceptional subclasses, where the inheritance of default purely probabilistic properties depends on the strength λ. However, z_λ-entailment does not inherit properties from classes to subclasses that are exceptional relative to some other property (and thus, like its classical counterpart, shows the problem of *inheritance blocking*).

Example 3.3 Let $KB = (L, P)$ be as in Example 2.1. Some tight z_λ-consequences, where $\lambda \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$, are shown in Table 2. Observe that, in contrast to Table 1, there are no empty intervals "$[1, 0]$", that is, no local inconsistencies. Then, observe that the default logical property of having legs is inherited from the class of birds down to yellow birds, independently from λ, while the default purely probabilistic property of being easy to see with a probability between 0.8 and 0.9 is also inherited from the class of yellow objects to yellow birds, but this is controlled by λ. Furthermore, for every strength $\lambda > 1/9$, these properties are not inherited down to the exceptional classes of penguins and yellow penguins, respectively. Note that for every strength $\lambda \leqslant 1/9$, the default logical property of having legs is inherited down to penguins, since there is only some weak inheritance of default purely probabilistic knowledge, and thus no conflict between the abilities to fly of birds and penguins.

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$legs\vert bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\vert yellow\wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\vert penguin$	$[1,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$legs\vert yellow\wedge penguin$	$[1,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$fly\vert bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly\vert yellow\wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly\vert penguin$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$
$fly\vert yellow\wedge penguin$	$[0,1]$	$[0,.5]$	$[0,.25]$	$[0,.17]$	$[0,.13]$	$[0,.1]$
$see\vert yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see\vert yellow\wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see\vert yellow\wedge penguin$	$[0,1]$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$

Table 2: Some tight z_λ-consequences.

Lexicographic Entailment of Strength λ. We finally recall the notion of lex_λ-entailment for λ-consistent $KB = (L, P)$. Note that, like z_λ-entailment, it can also be given in terms of a unique probability ranking for KB.

We use the z_λ-partition $(P_0, \ldots, P_k)$ of KB to define a lexicographic preference relation on probabilistic interpretations as follows. For probabilistic interpretations Pr and Pr', we say Pr is lex_λ***-preferable*** to Pr' iff some $i \in \{0, \ldots, k\}$ exists such that $|\{C \in P_i | Pr \models C\}| > |\{C \in P_i \mid Pr' \models C\}|$ and $|\{C \in P_j \mid Pr \models C\}| = |\{C \in P_j \mid Pr' \models C\}|$ for all $i < j \leqslant k$. A model Pr of a set of logical constraints and probabilistic formulas $\mathcal{F}$ is a lex_λ***-minimal model*** of $\mathcal{F}$ iff no model of $\mathcal{F}$ is lex_λ-preferable to Pr.

We now define the notion of lex_λ***-entailment*** as follows. A conditional constraint $C = (\psi|\phi)[l, u]$ is a lex_λ***-consequence*** of KB, denoted $KB \mathrel{|\!\!\sim}^{lex_\lambda} C$, iff every lex_λ-minimal model of $L \cup \{\phi \succcurlyeq \lambda\}$ satisfies C. It is a ***tight*** lex_λ***-consequence*** of KB, denoted $KB \mathrel{|\!\!\sim}^{lex_\lambda}_{tight} C$, iff $l = \inf Pr(\psi|\phi)$ (resp., $u = \sup Pr(\psi|\phi)$) subject to all lex_λ-minimal models Pr of $L \cup \{\phi \succcurlyeq \lambda\}$.

The following example shows that the notion of lex_λ-entailment realizes an inheritance of default properties, without showing the problem of inheritance blocking.

Example 3.4 Let $KB = (L, P)$ be as in Example 2.1. Some tight lex_λ-consequences, where $\lambda \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$, are shown in Table 3. In particular, for every strength $\lambda \in [0, 1]$, the default logical property of having legs is inherited from the class of birds to the exceptional subclass of penguins, while the default purely probabilistic property of being easy to see with a probability between 0.8 and 0.9 is also inherited from the class of yellow objects to the exceptional subclass of yellow penguins.

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$legs\vert bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\vert yellow\wedge bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\vert penguin$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs\vert yellow\wedge penguin$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$fly\vert bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly\vert yellow\wedge bird$	$[0,1]$	$[.5,1]$	$[.75,1]$	$[.83,1]$	$[.88,1]$	$[.9,1]$
$fly\vert penguin$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$
$fly\vert yellow\wedge penguin$	$[0,1]$	$[0,.5]$	$[0,.25]$	$[0,.17]$	$[0,.13]$	$[0,.1]$
$see\vert yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see\vert yellow\wedge bird$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$
$see\vert yellow\wedge penguin$	$[0,1]$	$[0,1]$	$[.5,1]$	$[.67,1]$	$[.75,1]$	$[.8,.9]$

Table 3: Some tight lex_λ-consequences.

4 Semantic Properties

In this section, we briefly summarize some semantic properties of λ-logical, z_λ-, and lex_λ-entailment.

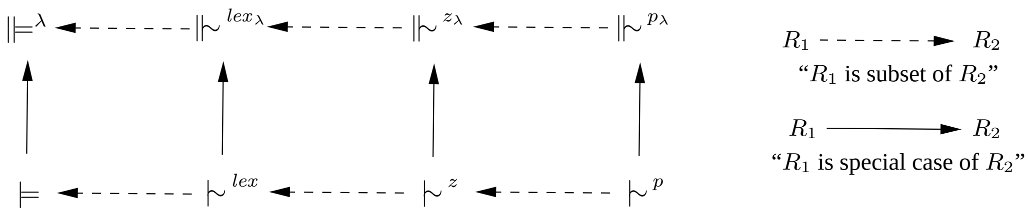

Figure 1: Relationships between probabilistic and ordinary formalisms.

General Nonmonotonic Properties. The notions of λ-logical, z_λ-, and lex_λ-entailment all satisfy probabilistic versions of the postulates *Right Weakening*, *Reflexivity*, *Left Logical Equivalence*, *Cut*, *Cautious Monotonicity*, and *Or* proposed by Kraus et al. [22], which are commonly regarded as being particularly desirable for any reasonable nonmonotonic entailment [31]. All three notions also satisfy the desirable property of *Rational Monotonicity* [22], which describes a restricted form of monotony and allows to ignore certain kinds of irrelevant knowledge.

Relationships between Probabilistic Formalisms. As for the relationships between the three formalisms, it holds that λ-logical entailment is stronger than both lex_λ- and z_λ-entailment. Moreover, lex_λ-entailment is stronger than z_λ-entailment. These relationships between λ-logical, z_λ-, and lex_λ-entailment are illustrated in Fig. 1.

In general, λ-logical entailment is strictly stronger than lex_λ-entailment, which in turn is strictly stronger than z_λ-entailment. However, in the special case when $\phi = \top$, the three notions of λ-logical, z_λ-, and lex_λ-entailment of $(\psi|\phi)[l,u]$ from λ-consistent $KB = (L,P)$ all coincide. Furthermore, also when $L \cup P \cup \{\phi \succcurlyeq \lambda\}$ is satisfiable, the three notions of λ-logical, z_λ-, and lex_λ-entailment of $(\psi|\phi)[l,u]$ from λ-consistent $KB = (L,P)$ all coincide.

Probabilistic and Classical Special Cases. For $\lambda = 0$, the notion of λ-logical entailment from KB coincides with standard logical entailment from KB. For $\lambda = 0$ (resp., $\lambda = 1$), the notions of z_λ- and lex_λ-entailment coincide with weak (resp., strong) probabilistic z- and lex-entailment introduced in [32] (resp., [30]). Furthermore, for $\lambda = 0$, the notion of λ-consistency coincides with the notion of g-coherence (see, e.g., [9]).

As for classical special cases, z_λ- and lex_λ-entailment of $(\beta|\alpha)[1,1]$ from λ-consistent probabilistic knowledge bases of the form $KB = (L,P)$, where $P = \{(\psi_i|\phi_i)[1,1] \mid i \in \{1,\ldots,n\}\}$, coincide with the classical notions of Pearl's entailment in System Z and Lehmann's lexicographic entailment of the default $\beta \leftarrow \alpha$ from the default counterpart of KB. Furthermore, λ-logical entailment of $(\beta|\alpha)[1,1]$ from such KB coincides with propositional logical entailment of $\beta \Leftarrow \alpha$ from the propositional counterpart of KB (see Fig. 1). Finally, the notion of λ-consistency for such KB coincides with the notion of ε- (or also p-) consistency for the default counterpart of KB.

Relationship to G-Coherent Entailment. Similarly to z_λ- and lex_λ-entailment, one can also define a probabilistic generalization of entailment in System P of strength $\lambda \in [0,1]$, called p_λ-*entailment*, which is strictly weaker than z_λ-entailment (see Fig. 1). However, since entailment in System P does not realize a general property inheritance along subclass relationships, also p_λ-entailment does not have such an inheritance, and in particular generally does not depend on λ (see Table 4, which shows some tight p_λ-consequences from KB of Example 2.1). For $\lambda = 0$, this notion of p_λ-entailment coincides with the notion of g-coherent entailment (see, e.g., [9]).

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$legs \mid bird$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
$legs \mid yellow \wedge bird$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$legs \mid penguin$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$legs \mid yellow \wedge penguin$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$fly \mid bird$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$	$[.9,1]$
$fly \mid yellow \wedge bird$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$fly \mid penguin$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$	$[0,.1]$
$fly \mid yellow \wedge penguin$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$see \mid yellow$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$	$[.8,.9]$
$see \mid yellow \wedge bird$	$[0,1]$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$see \mid yellow \wedge penguin$	$[0,1]$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$

Table 4: Some tight p_λ-consequences.

5 Further Examples

In reasoning from statistical knowledge and degrees of belief, z_1- and lex_1-entailment show a similar behavior as reference-class reasoning [23, 36] in a number of uncontroversial examples. But, they also avoid many drawbacks of reference-class reasoning. In detail, they can handle complex scenarios and even purely probabilistic subjective knowledge as input. Moreover, conclusions are drawn in a global way from all the available knowledge as a whole. See [30] for further details. The following example illustrates the use of lex_1-entailment for reasoning from statistical knowledge and degrees of belief.

Example 5.1 Suppose that we have the statistical knowledge "all penguins are birds", "between 90% and 95% of all birds fly", "at most 5% of all penguins fly", and "at least 95% of all yellow objects are easy to see". Moreover, assume that we believe "Sam is a yellow penguin". What do we then conclude about Sam's property of being easy to see? Under reference-class reasoning, which is a machinery for dealing with statistical knowledge and degrees of belief, we conclude "Sam is easy to see with a probability of at least 0.95". This is exactly what we obtain using the notion of lex_1-entailment. The above statistical knowledge can be represented by the probabilistic knowledge base $KB = (L, P)$, where $L = \{bird \Leftarrow penguin\}$ and $P = \{(fly|bird)[.9, .95], (fly|penguin)[0, .05], (see|yellow)[.95, 1]\}$. It is then not difficult to verify that KB is 1-consistent, and that $(see|yellow \wedge penguin)[0.95, 1]$ is a tight conclusion from KB under lex_1-entailment. Some other tight intervals for $see|yellow \wedge penguin$ from KB under λ-logical, z_λ-, lex_λ-, and p_λ-entailment are shown in Table 5.

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$\Vdash^{\lambda}_{tight}$	$\mathbf{[0,1]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$	$\mathbf{[1,0]}$
$\Vdash\!\sim^{lex_\lambda}_{tight}$	$\mathbf{[0,1]}$	$\mathbf{[.75,1]}$	$\mathbf{[.88,1]}$	$\mathbf{[.92,1]}$	$\mathbf{[.94,1]}$	$[.95,1]$
$\Vdash\!\sim^{z_\lambda}_{tight}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$
$\Vdash\!\sim^{p_\lambda}_{tight}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$

Table 5: Tight intervals for $see|yellow \wedge penguin$.

The next example is from the area of medical diagnosis.

Example 5.2 In a hospital, physicians have to diagnose whether patients with acute abdominal pain are suffering from appendicitis or not. Diagnosing appendicitis is a difficult task, since a lot of different symptoms (as, e.g., high temperature, a high rate of leucocytes, vomiting, and various types of pains) can indicate appendicitis, but often only the joint occurrence of several of these symptoms reliably supports the diagnosis. Here, we only consider four possible symptoms of appendicitis (*app*), namely a high rate of leucocytes (*leuco_high*) and the following three different types of pain: rectal pain (*rec_pain*), pain when released (*pain_rel*), and rebound tenderness (*reb_tender*). Thus, our view on this area is a very simplified one. Let our knowledge about the relationships between *app*, *leuco_high*, and the three types of pain be expressed by the following probabilistic knowledge base $KB = (L, P)$, where $L = \emptyset$ and P is given as follows:

$$
\begin{aligned}
P \;=\; \{ & (\mathit{reb\_tender} \mid \mathit{pain\_rel})[.70, .75], \\
& (\mathit{reb\_tender} \mid \mathit{leuco\_high})[.70, .75], \\
& (\mathit{app} \mid \mathit{rec\_pain} \wedge \mathit{pain\_rel})[.70, .75], \\
& (\mathit{app} \mid \mathit{rec\_pain} \wedge \mathit{reb\_tender})[.65, .70], \\
& (\mathit{app} \mid \mathit{pain\_rel} \wedge \mathit{reb\_tender} \wedge \mathit{leuco\_high})[.80, .85]\}\,.
\end{aligned}
$$

Suppose now that Judy is a patient showing the symptoms *leuco_high* and *pain_rel*. Which is the probability that Judy has appendicitis? Which is the probability that she has appendicitis given that she also feels rectal pain? Some tight intervals for $app|leuco\_high \wedge pain\_rel$ and $app|leuco\_high \wedge pain\_rel \wedge rec\_pain$, respectively, from KB under λ-logical, z_λ-, lex_λ-, and p_λ-entailment are shown in Tables 6 and 7, respectively.

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$\Vdash^{\lambda}_{tight}$	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$
$\Vdash\!\sim^{lex_\lambda}_{tight}$	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$
$\Vdash\!\sim^{z_\lambda}_{tight}$	$[0,1]$	$[.08,.99]$	$[.38,.93]$	$[.48,.91]$	$[.53,.9]$	$[.56,.9]$
$\Vdash\!\sim^{p_\lambda}_{tight}$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$

Table 6: Tight intervals for *app*|*leuco_high* ∧ *pain_rel*.

	Strength λ					
	0	0.2	0.4	0.6	0.8	1
$\Vdash^{\lambda}_{tight}$	$[0,1]$	$[0,1]$	$[.41,1]$	$[.57,1]$	$[.66,.92]$	$\mathbf{[1,0]}$
$\Vdash\!\sim^{lex_\lambda}_{tight}$	$[0,1]$	$[0,1]$	$[.41,1]$	$[.57,1]$	$[.66,.92]$	$[.7,.75]$
$\Vdash\!\sim^{z_\lambda}_{tight}$	$[0,1]$	$[0,1]$	$[.41,1]$	$[.57,1]$	$[.66,.92]$	$\mathbf{[0,1]}$
$\Vdash\!\sim^{p_\lambda}_{tight}$	$[0,1]$	$[0,1]$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$	$\mathbf{[0,1]}$

Table 7: Tight intervals for *app*|*leuco_high*∧*pain_rel*∧*rec_pain*.

6 Algorithms

Algorithm consistency in Fig. 3 decides whether a given probabilistic knowledge base $KB = (L, P)$ is λ-consistent. If KB is λ-consistent, then consistency also returns the z_λ-partition of KB. It is similar to an algorithm for deciding g-coherence by Biazzo et al. [8], which in turn is a probabilistic generalization of an algorithm for deciding ε-consistency in default reasoning [20]. More precisely, if $P = \emptyset$, then Step 1 returns the empty partition $()$, if L is satisfiable; and nil, otherwise. If $P \neq \emptyset$, then Steps 2-8 try to compute the z_λ-partition $\mathcal{P}$ of KB, and Step 9 returns $\mathcal{P}$, if this succeeds; and nil, otherwise.

Algorithms tight-s-consequence, with $s = \mathsf{z}$ and $s = \mathsf{lex}$, in Figs. 4 and 5 compute tight intervals under z_λ- and lex_λ-entailment from $KB = (L, P)$, respectively. They are similar to algorithms from [32] for computing tight entailed intervals under weak probabilistic z- and lex-entailment, respectively. Similarly to algorithms for lexicographic inference in [6], they are based on a compilation step. More precisely, if KB is not λ-consistent, then $[1, 0]$ is returned in Step 2. If KB is λ-consistent, and $L \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable, then $[1, 0]$ is returned in Step 4. Otherwise (that is, KB has a z_λ-partition $(P_0, \ldots, P_k)$, and $L \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable), we use the following Theorem 6.1 saying that then a set $\mathcal{D}^s_\alpha(KB) \subseteq 2^P$, $s \in \{z_\lambda, lex_\lambda\}$, exists such that $KB \Vdash\!\sim^s (\beta|\alpha)[l, u]$ iff $L \cup H \cup \{\alpha \succcurlyeq \lambda\} \Vdash (\beta|\alpha)[l, u]$ for all $H \in \mathcal{D}^s_\alpha(KB)$. In this case, we compute $\mathcal{D}^s_\alpha(KB)$ along the z_λ-partition of KB in Steps 5–9 (resp., 5–17), and the requested tight interval in Step 10 (resp., 18–22).

For $G, H \subseteq P$, we say G is z_λ-*preferable* to H iff some $i \in \{0, \ldots, k\}$ exists such that $P_i \subseteq G$, $P_i \not\subseteq H$, and $P_j \subseteq$

G and $P_j \subseteq H$ for all $i < j \leqslant k$. We say G is lex_λ*-preferable* to H iff some $i \in \{0, \ldots, k\}$ exists such that $|G \cap P_i| > |H \cap P_i|$ and $|G \cap P_j| = |H \cap P_j|$ for all $i < j \leqslant k$. For $\mathcal{D} \subseteq 2^P$ and $s \in \{z_\lambda, lex_\lambda\}$, we say G is s*-minimal* in $\mathcal{D}$ iff $G \in \mathcal{D}$ and no $H \in \mathcal{D}$ is s-preferable to G.

Theorem 6.1 *Let $KB = (L, P)$ be λ-consistent, $\beta|\alpha$ be a conditional event, and $L \cup \{\alpha \succcurlyeq \lambda\}$ be satisfiable. Let $s \in \{z_\lambda, lex_\lambda\}$ and $\mathcal{D}^s_\alpha(KB)$ be the set of all s-minimal elements in $\{H \subseteq P \mid L \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable$\}$. Then, l (resp., u) such that $KB \,\|\!\!\sim^s_{tight} (\beta|\alpha)[l, u]$ is given by $l = \min\, c$ (resp., $u = \max\, d$) subject to $L \cup H \cup \{\alpha \succcurlyeq \lambda\} \,\|\!\!\models_{tight} (\beta|\alpha)[c, d]$ and $H \in \mathcal{D}^s_\alpha(KB)$.*

The above three algorithms are based on reductions to (i) deciding whether a given $KB = (L, P)$ has a model Pr such that $Pr(\alpha) > 0$ for a given event α, and to (ii) computing tight logically entailed intervals from a given KB for a given conditional event $\beta|\alpha$. The number of tasks (i) and (ii) to be solved in the first two algorithms (resp., the third algorithm) is in $O(|P|^2)$ (resp., $O(2^{|P|})$). The task (i) can be reduced to deciding whether a system of linear constraints is solvable, while (ii) can be reduced to computing the optimal values of two linear programs. These two well-known results are summarized as follows.

Theorem 6.2 *Let $KB = (L, P)$ be a probabilistic knowledge base, and α, β be events. Let $R = \{I \in I_\Phi \mid I \models L\}$. Let LC denote the system of linear constraints in Fig. 2 over the variables y_r ($r \in R$). Then, (a) $L \cup P$ has a model Pr such that $Pr(\alpha) > 0$ iff LC is solvable. (b) If $L \cup P$ has a model Pr such that $Pr(\alpha) > 0$, then l (resp., u) such that $L \cup P \,\|\!\!\models_{tight} (\beta|\alpha)[l, u]$ is the optimal value of the following linear program over the variables y_r ($r \in R$):*

$$\textit{minimize (resp., maximize)} \sum_{\substack{r \in R \\ r \models \beta \wedge \alpha}} y_r \quad \textit{subject to LC}.$$

$$\begin{aligned}
\sum_{\substack{r \in R \\ r \models \neg\psi \wedge \phi}} -l\, y_r + \sum_{\substack{r \in R \\ r \models \psi \wedge \phi}} (1-l)\, y_r &\geqslant 0 \quad \text{(for all } (\psi|\phi)[l,u] \in P,\ l > 0)\\
\sum_{\substack{r \in R \\ r \models \neg\psi \wedge \phi}} u\, y_r + \sum_{\substack{r \in R \\ r \models \psi \wedge \phi}} (u-1)\, y_r &\geqslant 0 \quad \text{(for all } (\psi|\phi)[l,u] \in P,\ u < 1)\\
\sum_{\substack{r \in R \\ r \models \alpha}} y_r &= 1\\
y_r &\geqslant 0 \quad \text{(for all } r \in R)
\end{aligned}$$

Figure 2: System of linear constraints.

7 The System NMPROBLOG

The system NMPROBLOG is written in the programming language C, and uses "lp_solve 5.1" for deciding the solvability of systems of linear constraints and for computing the optimal values of linear programs. The graphical user interface (GUI) of NMPROBLOG has been built using "glade 2.6". Its main components are the main window, one window each for checking satisfiability, for checking λ-consistency, and for computing the z_λ-partition (see

Algorithm consistency

Input: probabilistic knowledge base $KB = (L, P)$ and strength λ.
Output: z_λ-partition of KB, if KB is λ-consistent; *nil* otherwise.

if $P = \emptyset$ **then if** L is satisfiable **then return** $()$ **else return** *nil*;
$R := P$;
$i := -1$;
repeat
  $i := i + 1$;
  $D[i] := \{(\psi|\phi)[l, u] \in R \mid L \cup R \cup \{\phi \succcurlyeq \lambda\}$ is satisfiable$\}$;
  $R := R \setminus D[i]$
until $R = \emptyset$ **or** $D[i] = \emptyset$;
if $R = \emptyset$ **then return** $(D[0], \ldots, D[i])$ **else return** *nil*.

Figure 3: Algorithm consistency

Algorithm tight-z-consequence

Input: probabilistic knowledge base $KB = (L, P)$,
conditional event $\beta|\alpha$, and strength λ.
Output: interval $[l, u] \subseteq [0, 1]$ such that $KB \,\|\!\!\sim^{z_\lambda}_{tight} (\beta|\alpha)[l, u]$.

$\mathcal{P} := \mathsf{consistency}(KB, \lambda)$;
if $\mathcal{P} = nil$ **then return** $[1, 0]$;
$(P_0, \ldots, P_k) := \mathcal{P}$; $R := L$;
if $R \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable **then return** $[1, 0]$;
$j := k$;
while $j \geqslant 0$ **and** $R \cup P_j \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable **do begin**
  $R := R \cup P_j$;
  $j := j - 1$
end;
compute $l, u \in [0, 1]$ s. t. $R \cup \{\alpha \succcurlyeq \lambda\} \,\|\!\!\models_{tight} (\beta|\alpha)[l, u]$;
return $[l, u]$.

Figure 4: Algorithm tight-z-consequence

Algorithm tight-lex-consequence

Input: probabilistic knowledge base $KB = (L, P)$,
conditional event $\beta|\alpha$, and strength λ.
Output: interval $[l, u] \subseteq [0, 1]$ such that $KB \,\|\!\!\sim^{lex_\lambda}_{tight} (\beta|\alpha)[l, u]$.

$\mathcal{P} := \mathsf{consistency}(KB, \lambda)$;
if $\mathcal{P} = nil$ **then return** $[1, 0]$;
$(P_0, \ldots, P_k) := \mathcal{P}$; $R := L$;
if $R \cup \{\alpha \succcurlyeq \lambda\}$ is unsatisfiable **then return** $[1, 0]$;
$\mathcal{H} := \{\emptyset\}$;
for $j := k$ **downto** 0 **do begin**
  $n := 0$;
  $\mathcal{H}' := \emptyset$;
  for each $G \subseteq P_j$ and $H \in \mathcal{H}$ **do**
    if $R \cup G \cup H \cup \{\alpha \succcurlyeq \lambda\}$ is satisfiable **then**
      if $n = |G|$ **then** $\mathcal{H}' := \mathcal{H}' \cup \{G \cup H\}$
        else if $n < |G|$ **then begin**
          $\mathcal{H}' := \{G \cup H\}$;
          $n := |G|$
        end;
  $\mathcal{H} := \mathcal{H}'$;
end;
$(l, u) := (1, 0)$;
for each $H \in \mathcal{H}$ **do begin**
  compute $c, d \in [0, 1]$ s. t. $R \cup H \cup \{\alpha \succcurlyeq \lambda\} \,\|\!\!\models_{tight} (\beta|\alpha)[c, d]$;
  $(l, u) := (\min(l, c), \max(u, d))$
end;
return $[l, u]$.

Figure 5: Algorithm tight-lex-consequence

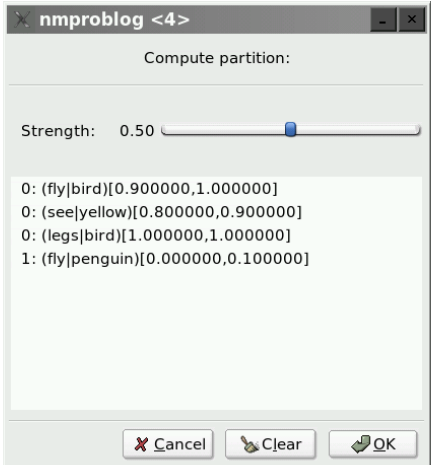

Figure 6: Window for computing the z_λ-partition.

Figure 7: Window for computing tight entailed intervals.

Fig. 6), as well as one window for computing tight entailed intervals (*result*) for any conditional event (*query*) under λ-logical, z_λ-, lex_λ-, and p_λ-entailment (see Fig. 7), for any *strength* $\lambda \in \{i/100 \,|\, i \in \{0,\ldots,100\}\}$.

The system NMPROBLOG loads from a file with suffix ".tax" a set of statements of one of the following forms: (i) $p=1$, where p is a nonempty string, to declare p as $\top$, (ii) $p=0$, where p is a nonempty string, to declare p as $\bot$, (iii) $p<1$, where p is a nonempty string, to declare p as a basic event, and (iv) $\psi > \phi$, where ψ and ϕ are events (in which "^", "&", and "#" encode $\neg$, $\wedge$, and $\vee$, respectively), to express that ϕ implies ψ. Furthermore, it then loads from a file with suffix ".prb" a set of statements of the form "ψ ϕ l u", where ψ and ϕ are events as above, and l and u are real numbers, to encode the conditional constraint $(\psi|\phi)[l,u]$. Note that every basic event in the ".prb"-file and in queries (window for computing tight entailed intervals) must be declared in the ".tax"-file.

Example 7.1 Consider again the probabilistic knowledge base $KB=(L,P)$ given in Example 2.1. The ".tax"-file contains the statements $1>bird$, $1>penguin$, $1>fly$, $1>legs$, $1>see$, $1>yellow$, and $bird>penguin$ to declare the basic events in KB and to express the logical constraints in L. The ".prb"-file contains the statements $legs$ $bird$ 1.0 1.0, see $yellow$ 0.8 0.9, fly $bird$ 0.9 1.0, and fly $penguin$ 0.0 0.1 to express the conditional constraints in P. After reading the ".tax"- and the ".prb"-file, one can open the window for computing tight consequences in Fig. 7 and, for example, compute the tight interval $[l,u]$ such that $KB \,\|\!\!\sim^{lex_\lambda}_{tight} (see|yellow \wedge bird)[l,u]$, $\lambda=0.5$, which is given by $[l,u]=[0.6,1]$ (see Fig. 7).

Example 7.2 Fig. 8 shows the time used by NMPROBLOG on a chain of n correlated basic events (2^n variables and $4(n-1)+1$ constraints in the generated linear optimization problems) for checking satisfiability and λ-consistency, as well as computing the z_λ-partition and tight entailed intervals under λ-logical, z_λ-, lex_λ-, and p_λ-entailment. Here, all the above reasoning tasks can be solved in few minutes, even when large linear optimization problems are generated (up to 16384 variables and 53 linear constraints).

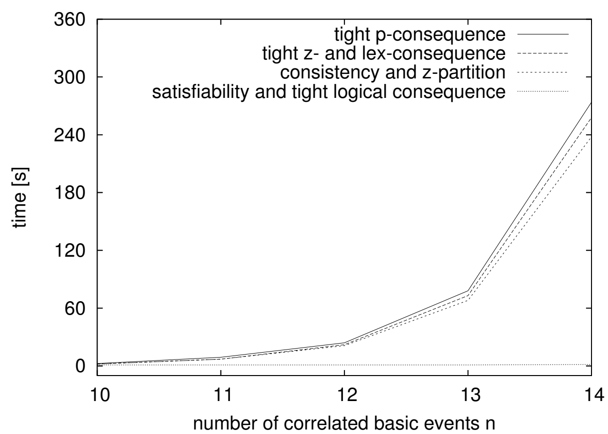

Figure 8: Time used by NMPROBLOG for different reasoning tasks on a chain of n correlated basic events (2^n worlds).

8 Summary and Outlook

I have recalled nonmonotonic probabilistic logics under variable-strength inheritance with overriding, namely, the notions of z_λ- and lex_λ-entailment, along with their semantic properties and some new examples. I have presented algorithms for deciding λ-consistency and for computing tight entailed intervals under z_λ- and lex_λ-entailment, which are based on reductions to the problems of deciding satisfiability and of computing tight logically entailed intervals in model-theoretic probabilistic logic.

Furthermore, I have presented the system NMPROBLOG, which comprises an implementation of these algorithms, and which is available at `http://www.kr.tuwien.ac.at/staff/lukasiew/nmproblog.tar.gz`. NMPROBLOG allows for (i) checking the satisfiability of probabilistic knowledge bases KB, (ii) checking the λ-consistency of KB, and (iii) computing the z_λ-partition of KB, as well as (iv) computing tight entailed intervals under any among λ-logical, lex_λ-, z_λ-, and p_λ-entailment, for any *strength* $\lambda \in \{i/100 \,|\, i \in \{0, \ldots, 100\}\}$. In particular, it thus also allows for probabilistic and default reasoning in all the special cases of λ-logical, lex_λ-, z_λ-, and p_λ-entailment that are summarized in Section 4.

A topic of future research is to explore whether there are techniques for more efficient or even tractable inference in nonmonotonic probabilistic logics under variable-strength inheritance with overriding (e.g., along the lines of [8] and [13]), and to eventually include them into NMPROBLOG.

Acknowledgments. This work has been supported by a Heisenberg Professorship of the German Research Foundation. I am thankful to the reviewers for their constructive comments, which helped to improve this work.

References

[1] E. W. Adams. *The Logic of Conditionals*, volume 86 of *Synthese Library*. D. Reidel, Dordrecht, Netherlands, 1975.

[2] F. Bacchus, A. Grove, J. Y. Halpern, and D. Koller. From statistical knowledge bases to degrees of belief. *Artif. Intell.*, 87(1-2):75–143, 1996.

[3] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proceedings IJCAI-1993*, pp. 640–645.

[4] S. Benferhat, D. Dubois, and H. Prade. Representing default rules in possibilistic logic. In *Proceedings KR-1992*, pp. 673–684.

[5] S. Benferhat, D. Dubois, and H. Prade. Nonmonotonic reasoning, conditional objects and possibility theory. *Artif. Intell.*, 92(1-2):259–276, 1997.

[6] S. Benferhat, S. Kaci, D. Le Berre, M.-A. Williams. Weakening conflicting information for iterated revision and knowledge integration. In *Proc. IJCAI-2001*, pp. 109–118.

[7] V. Biazzo and A. Gilio. A generalization of the fundamental theorem of de Finetti for imprecise conditional probability assessments. *Int. J. Approx. Reasoning*, 24:251–272, 2000.

[8] V. Biazzo, A. Gilio, T. Lukasiewicz, and G. Sanfilippo. Probabilistic logic under coherence: Complexity and algorithms. In *Proceedings ISIPTA-2001*, pp. 51–61.

[9] V. Biazzo, A. Gilio, T. Lukasiewicz, and G. Sanfilippo. Probabilistic logic under coherence, model-theoretic probabilistic logic, and default reasoning in System P. *Journal of Applied Non-Classical Logics*, 12(2):189–213, 2002.

[10] G. Boole. *An Investigation of the Laws of Thought, on which are Founded the Mathematical Theories of Logic and Probabilities*. Walton and Maberley, London, 1854. (Reprint: Dover Publications, New York, 1958).

[11] D. Dubois and H. Prade. On fuzzy syllogisms. *Computational Intelligence*, 4(2):171–179, 1988.

[12] D. Dubois, H. Prade, and J.-M. Touscas. Inference with imprecise numerical quantifiers. In Z. W. Ras and M. Zemankova, editors, *Intelligent Systems*, chapter 3, pp. 53–72. Ellis Horwood, 1990.

[13] T. Eiter and T. Lukasiewicz. Default reasoning from conditional knowledge bases: Complexity and tractable cases. *Artif. Intell.*, 124(2):169–241, 2000.

[14] R. Fagin, J. Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Inf. Comput.*, 87:78–128, 1990.

[15] N. Friedman and J. Y. Halpern. Plausibility measures and default reasoning. *J. ACM*, 48(4):648–685, 2001.

[16] A. M. Frisch and P. Haddawy. Anytime deduction for probabilistic logic. *Artif. Intell.*, 69:93–122, 1994.

[17] D. M. Gabbay and P. Smets, editors. *Handbook on Defeasible Reasoning and Uncertainty Management Systems*. Kluwer Academic, Dordrecht, Netherlands, 1998.

[18] A. Gilio. Probabilistic consistency of conditional probability bounds. In *Advances in Intelligent Computing, LNCS* 945, pp. 200–209. Springer, 1995.

[19] R. Giugno and T. Lukasiewicz. P-$\mathcal{SHOQ}(\mathbf{D})$: A probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$ for probabilistic ontologies in the Semantic Web. In *Proc. JELIA-2002*, pp. 86–97.

[20] M. Goldszmidt and J. Pearl. On the consistency of defeasible databases. *Artif. Intell.*, 52(2):121–149, 1991.

[21] M. Goldszmidt and J. Pearl. Qualitative probabilities for default reasoning, belief revision, and causal modeling. *Artif. Intell.*, 84(1-2):57–112, 1996.

[22] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artif. Intell.*, 14(1):167–207, 1990.

[23] H. E. Kyburg, Jr. *The Logical Foundations of Statistical Inference*. D. Reidel, Dordrecht, Netherlands, 1974.

[24] H. E. Kyburg, Jr. and C.-M. Teng. Evaluating defaults. In *Proceedings NMR-2002*, pp. 257–264.

[25] D. Lehmann and M. Magidor. What does a conditional knowledge base entail? *Artif. Intell.*, 55(1):1–60, 1992.

[26] D. Lehmann. Another perspective on default reasoning. *Ann. Math. Artif. Intell.*, 15(1):61–82, 1995.

[27] T. Lukasiewicz. Local probabilistic deduction from taxonomic and probabilistic knowledge-bases over conjunctive events. *Int. J. Approx. Reasoning*, 21(1):23–61, 1999.

[28] T. Lukasiewicz. Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res.*, 10:199–241, 1999.

[29] T. Lukasiewicz. Probabilistic logic programming under inheritance with overriding. In *Proc. UAI-2001*, pp. 329–336.

[30] T. Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Ann. Math. Artif. Intell.*, 34:35-88, 2002.

[31] T. Lukasiewicz. Nonmonotonic probabilistic reasoning under variable-strength inheritance with overriding. To appear in *Synthese*.

[32] T. Lukasiewicz. Weak nonmonotonic probabilistic logics. In *Proceedings KR-2004*, pp. 23–33. Extended version to appear in *Artif. Intell.*.

[33] N. J. Nilsson. Probabilistic logic. *Artif. Intell.*, 28(1):71–88, 1986.

[34] J. Pearl. Probabilistic semantics for nonmonotonic reasoning: A survey. In *Proceedings KR-1989*, pp. 505–516.

[35] J. Pearl. System Z: A natural ordering of defaults with tractable applications to default reasoning. In *Proceedings TARK-1990*, pp. 121–135.

[36] J. L. Pollock. *Nomic Probabilities and the Foundations of Induction*. Oxford University Press, Oxford, 1990.

[37] G. R. Wheeler: A resource bounded default logic. In *Proceedings NMR-2004*, pp. 416–422.